



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/616,525	07/10/2003	Mark Robert Funk	ROC920020205US1	1216
7590	11/20/2009		EXAMINER	
Grant A. Johnson IBM Corporation - Dept. 917 3605 Highway 52 North Rochester, MN 55901			RUTTEN, JAMES D	
			ART UNIT	PAPER NUMBER
			2192	
			MAIL DATE	DELIVERY MODE
			11/20/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte MARK ROBERT FUNK and MICHAEL JAMES DENNEY

Appeal 2009-005282
Application 10/616,525
Technology Center 2100

Decided: November 20, 2009

Before LANCE LEONARD BARRY, JEAN R. HOMERE, and THU A. DANG, *Administrative Patent Judges*.

DANG, *Administrative Patent Judge*.

DECISION ON APPEAL

Appellants appeal from the Examiner's final rejection of claims 1, 2, 4-11, 13 and 14 under 35 U.S.C. § 134 (2002). Claims 3 and 12 have been canceled. We have jurisdiction under 35 U.S.C. § 6(b) (2002).

We affirm.

I. STATEMENT OF THE CASE

A. INVENTION

According to Appellants, the invention relates generally to the data processing field, and more particularly, relates to implementing breakpoint based performance measurement (Spec. 1, ll. 5-7).

B. ILLUSTRATIVE CLAIM

Claim 1 is exemplary and is reproduced below:

1. A method for implementing breakpoint based performance measurement using a set of hardware counters for counting hardware events; said hardware counters being programmable for counting predefined programmable processor events; said predefined programmable processor events including processor cycles and cache misses; said method comprising:

providing compiler-generated hardware instructions defining breakpoint instructions within an instruction stream; said compiler-generated hardware instructions including a start breakpoint instruction and a stop breakpoint instruction;

inserting said start breakpoint instruction and said stop breakpoint instruction in compiler-generated hardware instructions for a user source code;

executing said compiler-generated hardware instructions and suspending processing of said hardware instructions responsive to executing said start breakpoint instruction;

responsive to executing said start breakpoint instruction generating a processor interrupt for entering interrupt handler instructions and calling breakpoint instructions;

said breakpoint instructions generating a start processing instruction to return processing from said interrupt handler instructions to said compiler-generated hardware instructions and starting said defined set of hardware counters, responsive to said generated start processing instruction;

 executing said compiler-generated hardware instructions and suspending processing of said compiler-generated hardware instructions and stopping said defined set of hardware counters, responsive to executing said stop breakpoint instruction; and

 providing a debugger breakpoint manager including a performance measurement program and a user interface, and enabling a user to specify a start bound and an end bound of a performance collection region of said user source code and said set of hardware counters.

C. REJECTIONS

The Examiner relied upon the following prior art in rejecting the claims on appeal:

Hawley	US 5,533,192	Jul. 02, 1996
Dreyer	US 5,657,253	Aug. 12, 1997
Carter	US 6,249,907 B1	Jun. 19, 2001

Bickle, *Differential Effective Lapse Time Accumulator (Delta)*, 27(2) IBM Technical Disclosure, July 1, 1984 at 1258-1259.

Rosenberg, *How Debugger Work: Algorithms, Data, Structures, and Architecture* 1-53 (John Wiley & Sons, Inc. 1996)

Appeal 2009-005282
Application 10/616,525

Claims 1, 2, 4-6, 8-11, 13, and 14 stand rejected under 35 U.S.C. § 103(a) over the teachings of Bickle in view of Rosenberg, Dreyer, and Carter.

Claim 7 stands rejected under 35 U.S.C. § 103(a) over the teachings of Bickle in view of Rosenberg, Dreyer, Carter, and Hawley.

II. ISSUE

Have Appellants shown that the Examiner erred in finding that the combination of Bickle, Rosenberg, Dreyer, and Carter teaches or would have suggested “inserting said start breakpoint instruction and said stop breakpoint instruction in compiler-generated hardware instructions for a user source code” and “said breakpoint instructions generating a start processing instruction to return processing from said interrupt handler instructions to said compiler-generated hardware instructions and starting said defined set of hardware counters, responsive to said generated start processing instruction” (claim 1)?

III. FINDINGS OF FACT

The following Findings of Fact (FF) are shown by a preponderance of the evidence.

Bickle

1. Bickle discloses accumulating elapsed time between start breakpoint A and stop breakpoint B (Bickle 1, ll. 25-29).

Rosenberg

2. Rosenberg discloses implementing breakpoints as a special instruction that causes a trap to the operating system (Rosenberg 40, ll. 23-25).
3. The operating system notifies a debugger that the debuggee stopped, why it stopped, and where it stopped, including which thread of execution was running when the stop occurred (Rosenberg 22, ll. 9-12).
4. The debugger re-inserts the saved instruction to proceed and replace the breakpoint instruction, wherein after breakpoint processing has completed and the debugger is instructed to continue execution of the debuggee. It replaces the original instruction, single-step over that single instruction, and then puts the breakpoint back before letting the debuggee proceed at full speed (Rosenberg 42, ll. 1-10).

Carter

5. Carter discloses a system for debugging a computer program using specified breakpoint type (Abstract), wherein executable object code is generated including the hook function calls inserted by the compiler (col. 5, ll. 31-37) and wherein a verb breakpoint command instructs the compiler to generate hook function calls (col. 6, ll. 44-48).

IV. PRINCIPLES OF LAW

The general allegation of patentability does not specify, as required, how the underlined language patentably distinguishes the claimed invention. This form of argument is wholly ineffective in demonstrating

Appeal 2009-005282
Application 10/616,525

error in the Examiner's *prima facie* case to establish the patentability of the claims on appeal. *Ex parte Belinne*, Appeal No. 2009-004693, decided Aug. 10, 2009, (BPAI) (informative). Available at:

<http://www.uspto.gov/web/offices/dcom/bpai/its/fd09004693.pdf>.

Section 103 forbids issuance of a patent when "the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains."

KSR Int'l Co. v. Teleflex Inc., 550 U.S. 398, 406 (2007).

The Supreme Court reaffirmed principles based on its precedent that "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." *Id.* at 416. The operative question is thus "whether the improvement is more than the predictable use of prior art elements according to their established functions." *Id.* at 417.

The Court noted that "[c]ommon sense teaches . . . that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* at 420. "A person of ordinary skill is also a person of ordinary creativity, not an automaton." *Id.* at 421.

In affirming a determination of obviousness, the Federal Circuit has relied, in part, on an applicant's failure to present evidence that the modification was "uniquely challenging or difficult for one of ordinary skill in the art" or "represented an unobvious step over the prior art." *Leapfrog*

Appeal 2009-005282
Application 10/616,525

Enter., Inc. v. Fisher-Price, Inc., 485 F.3d 1157, 1162 (Fed. Cir. 2007)(citing *KSR*, 550 U.S. at 418-19).

One cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. *In re Merck & Co.*, 800 F.2d 1091, 1097 (Fed. Cir. 1986) (citing *In re Keller*, 642 F.2d 413, 425 (CCPA 1981).

V. ANALYSIS

Claims 1, 2, 4-6, 8-11, 13, and 14

In this decision, we have considered only those arguments actually made by Appellants. Arguments which Appellants could have made but did not make in the Brief have not been considered and are deemed to be waived. *See* 37 C.F.R. § 41.37(c)(1)(vii).

On pages 25-28 of the Brief, Appellants assert that “[t]he Board should reverse the rejection” (App. Br. 25), merely repeat the language of claim 1 (App. Br. 25-27), and then contend that “Bickle, Rosenberg, Dreyer et al., Carter et al., and Howley references fail to teach” highlighted portions of the claim (App. Br. 27-28). However, the general allegation of patentability does not specify, as required, how the highlighted language of claim 1 patentably distinguishes the claimed invention over the applied references. This form of argument is wholly ineffective in demonstrating error in the Examiner’s *prima facie* case to establish the patentability of the claims on appeal. *See Ex parte Belinne*, Appeal No. 2009-004693. As the Examiner finds, “Appellants’ arguments generally fail to comply with 37

C.F.R. § 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references" (Ans. 10).

For similar reason, by mere repeating of the language of claim 1 on page 28, 30, 32-33, and 36-41, Appellants do not specify how the language of claim 1 patentably distinguishes the invention. We agree with the Examiner that such general allegations of patentability do not meet Appellants' burden (Ans. 10).

Appellants also contend that “[t]he present invention provides enhanced breakpoint based performance measurement that is different from the prior art” (App. Br. 27). However, such contention is not commensurate in scope with the language of claim 1 since claim 1 does not recite any such “enhanced” feature.

Appellants further contend that “Rosenberg does not teach any ‘special hardware instruction’, nor any equivalent hardware instruction” (App. Br. 29), and that “Carter does not disclose the claimed steps of the invention” (*Id.*). Appellants appear to be arguing that Rosenberg and Carter do not *individually* disclose the claimed invention. However, the Examiner has rejected the claims based on *the combination* of Bickle, Rosenberg, Dreyer, and Carter. It has been held that nonobviousness cannot be shown by attacking the references individually. *See In re Merck*, 800 at 1097.

The Examiner finds that “[s]ince these instructions [of Rosenberg] are used by the CPU, they are clearly interpreted as being hardware

instructions" (Ans. 12). The Examiner concludes that the claims are unpatentable over Bickle in view of Rosenberg, in view of Dreyer, in view of Carter (Ans. 3).

Thus, an issue we address on appeal is whether Appellants have shown that the Examiner erred in concluding that the combination of Bickle in view of Rosenberg, Dreyer, and Carter teaches or would have suggested "inserting said start breakpoint instruction and said stop breakpoint instruction in compiler-generated hardware instructions for a user source code" and "said breakpoint instructions generating a start processing instruction to return processing from said interrupt handler instructions to said compiler-generated hardware instructions and starting said defined set of hardware counters, responsive to said generated start processing instruction" (claim 1).

Bickle discloses accumulating elapsed time between start breakpoint A and stop breakpoint B (FF 1). We find an artisan would have understood Bickle to disclose a start breakpoint instruction and a stop breakpoint instruction.

Further, Rosenberg discloses implementing breakpoints as a special instruction that causes a trap to the operating system (FF 2), wherein the saved instruction is re-inserted after breakpoint processing has completed to continue execution (FF 3-4). We find the ordinarily skilled artisan would also have understood Rosenberg to disclose inserting breakpoint instructions and returning processing from said interrupt handler instructions.

Carter discloses a system for debugging a computer program using specified breakpoint type, wherein hook functions are generated and inserted by the compiler (FF 5). The ordinarily skilled artisan would have understood the hook functions of Carter to be compiler-generated hardware instructions.

Accordingly, we agree with the Examiner that the combination of Bickle, Rosenberg, Dreyer, and Carter teaches or would have suggested “inserting said start breakpoint instruction and said stop breakpoint instruction in compiler-generated hardware instructions for a user source code” and “said breakpoint instructions generating a start processing instruction to return processing from said interrupt handler instructions to said compiler-generated hardware instructions and starting said defined set of hardware counters, responsive to said generated start processing instruction” as required by claim 1.

Appellants contend that “[t]o arrive at such a conclusion [that Carter discloses the claimed steps] appears to require that the above expressly recited limitations set forth in claim 1 of the method of the invention are ignored or improperly incorporated applicants’ teachings into Carter” (App. Br. 29), and merely cite to numerous case laws without providing any explanation as to how the case laws are applicable (App. Br. 29-34.) Appellants appear to be arguing that there is no suggestion found in Carter to combine the references. However, the Examiner concludes that it would have been obvious “to use Carter’s hook functions with Bickle’s breakpoints

in order to enable a debugger to stop execution as suggested by Carter”
(Ans. 6-7).

We agree with the Examiner. Since Carter teaches the use of compiler-generated hardware instructions, we conclude that substitution of one known element (the instructions to be provided with breakpoint instructions) with another (the compiler-generated hardware instructions of Carter) would have yielded predictable results to one of ordinary skill in the art at the time of the invention. We find that providing compiler-generated hardware instructions as taught by Carter with breakpoint instructions is no more than a simple arrangement of old elements, with each performing the same function it had been known to perform, yielding no more than one would expect from such an arrangement. *See KSR*, 550 U.S. at 417.

Appellants have presented no evidence that using Carter’s compiler-generated hardware instructions in place of instructions (to be inserted with start breakpoint instruction and stop breakpoint instruction) as taught by Bickle, Rosenberg, and Dreyer was “uniquely challenging or difficult for one of ordinary skill in the art” or “represented an unobvious step over the prior art.” *Leapfrog*, 485 F.3d at 1162 (citing *KSR*, 550 U.S. at 418-19). Rather, Appellants’ claimed invention is simply an arrangement of the well-known teaching of breakpoint instructions, with the well-known teaching of compiler-generated hardware instructions. Thus, the combined teachings of the references represent merely a combination of familiar elements according to known methods and do no more than yield predictable results. *See KSR*, 550 U.S. at 416.

Thus, Appellants have neither shown that the Examiner failed to make a prima facie case of obviousness, nor have they persuasively rebutted the Examiner's prima facie case. Accordingly, we find that the Appellants have not shown that the Examiner erred in rejecting independent claim 1 and claims 2 and 5 depending therefrom and falling therewith under 35 U.S.C. § 103(a).

As to dependent claim 4, Appellants do not provide separate arguments but merely repeat the claim language, repeat that "no teaching or motivation exists," and again cite various case laws without providing any explanation as to how the case laws are applicable (App. Br. 35-36). As discussed above regarding claim 1, we conclude that the combination of Bickle, Rosenberg, Dreyer and Carter discloses the claimed invention. We also conclude that Appellants have neither shown that the Examiner failed to make a prima facie case of obviousness, nor have they persuasively rebutted the Examiner's prima facie case. Accordingly, we find that the Appellants have not shown that the Examiner erred in rejecting dependent claim 4 depending from claim 1 under 35 U.S.C. § 103(a).

Similarly, as to independent claims 6 and 11, Appellants do not provide separate arguments but merely repeat the claim language, repeat that "no teaching, suggestion, or motivation provided" and "no sufficient reason exists so that one skilled in the art would have arrived at the claimed subject matter" and again cite various case laws without providing any explanation as to how the case laws are applicable (App. Br. 36-43). As discussed above regarding claim 1, we conclude that the combination of Bickle, Rosenberg,

Dreyer and Carter discloses the claimed invention. We further conclude that Appellants have neither shown that the Examiner failed to make a *prima facie* case of obviousness, nor have they persuasively rebutted the Examiner's *prima facie* case.

Accordingly, we find that the Appellants also have not shown that the Examiner erred in rejecting independent claim 6 and 11, and claims 8-10, 13 and 14 respectively depending therefrom and falling therewith under 35 U.S.C. § 103(a).

Claim 7

Appellants do not provide separate arguments for claim 7 from the rejection of claim 6, from which claim 7 depends, but merely contend that "Bickle, Rosenberg, Dreyer, Carter and Hawley... fail to suggest" the language of claim 7 (App. Br. 44).

As discussed above, we find no deficiency regarding Bickle in view of Rosenberg, Dreyer, and Carter in the rejection of claim 1, and 6 falling therewith. Accordingly, we find that the Appellants have not met their burden of showing error in the rejection of claim 7 depending from claim 6 over the combination of Bickle, Rosenberg, Dreyer, Carter, and Hawley.

Further, the general allegation of patentability does not specify, as required, how the language of claim 7 patentably distinguishes the claimed invention over the applied references. We conclude that the Appellants have not shown that the Examiner erred in rejecting claim 7 under 35 U.S.C. § 103(a).

CONCLUSIONS OF LAW

- (1) Appellants have not shown that the Examiner erred in concluding that claims 1, 2, 4-6, 8-11, 13, and 14 are unpatentable over the combination of Bickle, Rosenberg, Dreyer, and Carter .
- (2) Appellants have not shown that the Examiner erred in concluding that claim 7 is unpatentable over the teachings of the combination of Bickle, Rosenberg, Dreyer, Carter, and Hawley.
- (3) Claims 1, 2, 4-11, 13, and 14 are not patentable.

DECISION

We affirm the Examiner's decision rejecting claims 1, 2, 4-11, 13, and 14 under 35 U.S.C. § 103(a).

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED

peb

GRANT A. JOHNSON
IBM CORPORATION - DEPT. 917
3605 HIGHWAY 52 NORTH
ROCHESTER, MN 55901